WHAT IS CLAIMED IS:

1. A method of sampling instructions executing in a multi-threaded processor comprising:

selecting an instruction for sampling;

storing information relating to the instruction;

determining whether the instruction includes an event of interest, the event of interest including information relating to a thread within which the instruction is executing; and

reporting the instruction if the instruction includes an event of interest on a per-thread basis.

2. The method of claim 1 further comprising

providing a register with a bit vector representing a plurality of events of interest; and wherein

the determining whether the instruction includes the event of interest further includes comparing the information relating to the instruction to the bit vector to determine whether the information relating to the instruction corresponds to a thread of interest.

3. The method of claim 1 wherein the comparing is via at least one of a mask operation or a more expressive operation.

4. The method of claim 1 wherein

the selecting the instruction is without regard to a thread to which the instruction is bound.

5. The method of claim 1 further comprising

identifying a thread to which the instruction is bound when the instruction is selected.

6. The method of claim 1 further comprising

providing filtering criteria on a per-thread basis.

7. The method of claim 1 further comprising
providing a single set of filtering criteria; and,
scheduling sampling among a plurality of threads via software.

8. A method of sampling instructions executing in a multi-threaded processor comprising:
setting a candidate counter to a number
selecting an instruction for sampling;
storing information relating to the instruction;
determining whether all events for the instruction have occurred;
decrementing the candidate counter when all events for the instruction have occurred and when the instruction corresponds to a desired sampled thread;
determining whether the candidate counter equals zero; and
reporting the instruction when the candidate counter equals zero.

9. The method of claim 8 wherein
the information relating to the instruction represents an instruction history, and
the instruction history includes information relating to at least one of an events value, a program counter value, a branch target address value, an effective memory address value, a latency value, a number in issue bundle value, a number in retire bundle value, a privilege value, a branch history value and a number in fetch bundle value.

10. The method of claim 8 wherein
the selecting the instruction is without regard to a thread to which the instruction is bound.

11. The method of claim 8 further comprising
identifying a thread to which the instruction is bound when the instruction is selected.

12. The method of claim 8 further comprising
providing filtering criteria on a per-thread basis.

13. The method of claim 8 further comprising
providing a single set of filtering criteria; and,
scheduling sampling among a plurality of threads via software.

14. A method of sampling instructions executing in a multi-threaded processor comprising:
setting a candidate counter to a number
selecting an instruction for sampling;
storing information relating to the instruction;
determining whether all events for the instruction have occurred;
determining whether the instruction includes events of interest, the events of interest including whether the instruction corresponds to a desired sampled thread;
decrementing the candidate counter when all events for the instruction have occurred and when the instruction includes events of interest;
determining whether the candidate counter equals zero; and
reporting the instruction when the candidate counter equals zero.

15. The method of claim 14 further comprising
providing a register with a bit vector representing events of interest; and
wherein
the determining whether the instruction includes events of interest further includes comparing the information relating to the instruction to the bit vector.

16. The method of claim 14 wherein
the information relating to the instruction represents an instruction history, and
the instruction history includes information relating to at least one of an event value, a program counter value, a branch target address value, an

effective memory address value, a latency value, a number in issue bundle value, a number in retire bundle value, a privileged value, a branch history value and a number in fetch bundle value.

17. The method of claim 14 wherein
the selecting an instruction for sampling is based upon sample selection criteria; and
the sample selection criteria include information relating to a desired sampled thread.

18. A sampling mechanism for sampling an instruction comprising:
sampling logic, the sampling logic determining whether the instruction corresponds to a desired sampled thread;
sampling register logic coupled to the sampling logic;
instruction history register logic coupled to the sampling register logic, the instruction history register logic storing information relating to the instruction; and,
sample filtering and counting logic coupled to the sampling logic.

19. The sampling mechanism of claim 18 further comprising:
notification logic, the notification logic reporting the information relating to the instruction if the instruction corresponds to the desired sampled thread.

20. The sampling mechanism of claim 18 wherein the sampling register logic includes
a register with a bit vector representing events of interest; and wherein
the sampling logic determines whether the instruction includes events of interest by comparing the information relating to the instruction to the bit vector.

21. The sampling mechanism of claim 18 wherein
the information relating to the instruction represents an instruction history, and

the instruction history includes information relating to at least one of an events value, a program counter value, a branch target address value, an effective memory address value, a latency value, a number in issue bundle value, a number in retire bundle value, a privileged value, a branch history value and a number in fetch bundle value.

22. The sampling mechanism of claim 18 wherein

the sampling register logic includes a sample selection criteria register storing sample selection criteria; and

the sample selection criteria include information relating to a desired sampled thread.